Journal of Optimization in Industrial Engineering
Vol.15, Issue 1, Winter & Spring 2022, 31-45
DOI:10.22094/JOIE.2021.1914607.1804

Research Article

An Integrated Bi-Objective Mathematical Model for Minimizing Take-off Delay and Passenger Dissatisfaction

Razieh Larizadeh[a], Reza Ramezanian[a,*]

[a] *Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran*

Received 14 November 2020; Revised 31 May 2021 ; Accepted 02 June 2021

Abstract

As air transportation has increased in recent years, it is necessary for airport planners to optimally manage aircraft ground traffic on stands, taxiways and runways in order to minimize flight delay and passenger dissatisfaction. A closer look at the literature in this area indicates that most studies have merely focused on one of these resources which in a macroscopic level may result in aircrafts' collision and ground traffic at the airport. In this paper, a new bi-objective Mixed-Integer Linear Programming (MILP) model is developed to help airport management to integrate Gate Assignment Problem (GAP) and Runway Scheduling Problem (RSP) considering taxiing operation for departing flights. The proposed model aims to help airport planners to 1) minimize any deviation from preferred schedule and 2) minimize transit passengers' walking distance. Due to the complexity of the research problem, a Normalized Weighted Sum Method (NWSM) is applied to solve small-sized problems and two meta-heuristics, namely NSGA-II and MOGWO, are used for large-scale instances to generate Pareto optimal solutions. The performance of these algorithms is assessed by well-known coverage and convergence measures. Based on the most criteria, the results indicate that MOGWO outperforms NSGA-II.

Keywords: Airport Planning; Airport Schedule; Flight Delays; NWSM; NSGA-II; MOGWO

1. Introduction

Take-off/landing operation is considered as a continuous chain of resource assignment and resource scheduling. With the rise of air travel in recent decades, the problem of resource planning has been of great importance to reduce flight delay and increase airport profitability. Considering all airport ground assets including gates, runways, and taxiways is crucial to make an integrated plan for aircraft movements in landside area.

Gate Assignment Problem (GAP) is defined as an issue of assigning a set of aircrafts to a set of gates and determining their sequence due to technical constraints such as impossibility of assigning more than one flight to one gate at the same time, limitation of some gates in servicing to the specific size of aircrafts, limitation of two adjacent gates in servicing to the flights that have same departure time, etc (Yu, 2015; Khakzar Bafruei et al., 2018). The most common objective function of GAP is to minimize passenger walking distance to/between gates. Other objective functions that have been commonly used in the literature include minimization of gate idle time, number of ungated flights, and flight waiting time (Aktel et al., 2017; Nourmohammadzadeh, 2012).

Another problem is to schedule runways which are usually considered as the system bottlenecks. Although a straightforward solution for this problem is to increase the number of runways, it is widely admitted that the effective use of current resources is more pragmatic than infrastructure development (Bennell et al., 2017). Runway Scheduling Problem (RSP) aims to assign an available runway to each aircraft (if there are several runways) and determine the optimal sequence of operations (Bennell et al., 2011). It is worth mentioning that the runway capacity of an airport is defined as the maximum possible rate of aircraft landings and take-offs which are supported by a single or multiple runway. Common objective functions in RSP include minimization of average tardiness, average completion time, completion time of the last operation (Makespan), and maximization of the runway throughput, etc. One of the most important constraints that has always been of interest on runway scheduling is Wake Vortex Effect (VWE). A wake vortex is a potentially hazardous effect that the rolling moment of a leading aircraft can impose on the any other aircraft behind. Therefore, the sequence-dependent WV separation time between consecutive aircrafts is necessary to be taken into account (**Lieder** and Stolletz, 2016). In this regard, International Civil Aviation Organization (ICAO) defined separation time standards between leader and follower aircrafts for approach, landing, and take-off to provide a safe condition at an airport area (See Table 1).

*Corresponding author Email address: ramezanian@kntu.ac.ir

Table 1
Minimum separation time (in seconds) required between consecutive aircrafts on the same runway (Lieder and Stolletz, 2016)

			Following aircraft					
			Landing			Take-off		
			Small	**Large**	**Heavy**	**Small**	**Large**	**Heavy**
Leading aircraft	**Landing**	**Small**	82	69	60	75	75	75
		Large	131	69	60	75	75	75
		Heavy	196	157	96	75	75	75
	Take-off	**Small**	60	60	60	60	60	60
		Large	60	60	60	60	60	60
		Heavy	60	60	60	120	120	90

It should be noted that aircraft movement from gate to runway is operated on a set of routes which is called taxi-network. Aircraft taxiing should be scheduled without any collision through the most efficient path. Hence, traffic controllers must consider the taxi routing problem as an indispensable link between GAP and RSP. The easiest way to determine an aircraft's taxiway is to route it on the shortest path. Such route is usually considered to be fixed and independent of any immediate changes in traffic conditions. Although this approach is easy to use, it does not provide the optimal solution since airport traffic is constantly changing (See Figure 1).

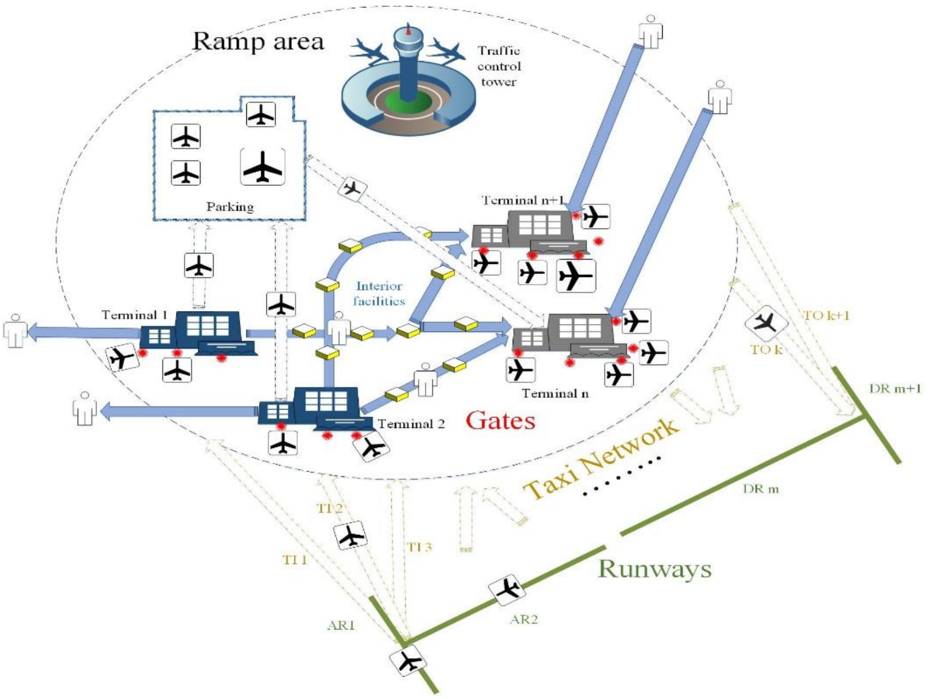

Fig. 1. Transfer flow at an airport

This paper formulates a MILP model in order to integrate gate assignment and runway scheduling while considering aircraft taxiing. The aim of the proposed model is to minimize two objective functions: 1) deviation of preferred scheduling, and 2) transit passenger dissatisfaction because of long walking distance between gates.

2. Literature Review

Given the sharp increase in air transportation over recent years, providing an optimal gate assignment plan is a crucial decision that traffic control tower faces every day. Lee et al. (2016) tried to determine departing and arriving passengers' gates at an airport using a single objective mixed integer programing model. They aimed to assign given aircrafts to available gates while keeping the passengers' flow balanced. They also performed a simulation experiment to verify the effect of their proposed model on the internal gate efficiency. The results indicated that the passenger processing time was reduced considerably. Aktel et al. (2017) tried to minimize both the number of flights assigned to the apron (ungated flights) and total passenger walking distance as a bi-objective MIP model. They proposed a new Tabu Search (TS) algorithm, a Simulated Annealing (SA) algorithm and a greedy algorithm to address the problem and find the optimal Pareto front. Khakzar Bafruei et al. (2018) stated that the gate processing time is not a fixed parameter in real-world planning and should be considered continuously controllable in a given interval. They proposed a bi-objective mixed-integer model to minimize 1) the total cost of delay and 2) the passenger overcrowding. The problem was solved by the implementation of a Multi-Objective Harmony Search Algorithm (MOHSA) and a Non-dominated Sorting Genetic Algorithm II (NSGA-II). In a recent research conducted by Das et al. (2020), the past studies on GAP reviewed to clarify various types of formulations, objectives, and solution methods .They indicated that there was not a standard problem formulation for GAP due to the multitude of stakeholders, feasibility requirements, and objective functions. However, they showed that the passenger-oriented objectives were most common among past research. In addition, they reported that most studies employed multi-objective functions and heuristic/metaheuristic approaches.

On the other hand, many studies have been focused on RSP which is widely known as the bottleneck of the whole air transportation system. Atkin et al. (2007) focused on RSP and proposed an automated advisory system to aid runway controllers obtain optimal take-off orders on a single departure runway. The aim was to reduce the total separations and increase runway throughput. They presented a hybrid meta-heuristic system that took account of more aircrafts than a human controller. In the real world, however, many congested airports have more than a single runway for arriving/departing aircrafts. Consequently, Messaoud et al. (2017) presented a MILP model for the aircraft landing problem on a pair of runways. However, they did not consider the WV effect on adjacent runways. They generated some small-sized problems and solved them using Lingo. Pohl et al. (2020) presented a single objective MILP model for multiple runway scheduling problem under the consideration of snow removal slots. They assumed that all runways could be used by both arrivals and departures without any limitation. To improve the computational tractability of Branch and Bound (B&B) algorithm, they developed pruning rules and valid inequalities. The computational results of this study indicated that their solution approach caused less delay cost compared to the previous common approaches.

According to the literature, most studies on GAP and RSP have been carried out as separate independent problems. Although rare, there are some studies which consider these two areas of airport scheduling together. Nourmohammadzadeh (2012) studied integrated GAP and RSP and presented a bi-objective mathematical model aiming to 1) minimize the total waiting time at both gate and runway and 2) minimize the passenger walking distance at the airport terminal area. He compared the performance of three metaheuristics (NSGA-II, Pareto Simulating Annealing (PSA) and a hybrid of NSGA-II and PSA) on generated large-scale problems and came to the conclusion that the proposed hybrid algorithm showed a better performance. He assumed a fixed and predetermined taxi time between gates and runways without considering the possibility of ground traffic. Guepet et al. (2016) presented a MIP model to formulate single path Ground Routing Problem (GRP) and generalized it to include alternative paths. They considered an objective function that included average delay and the percentage of flights having a delay less than 15 minutes. Clare and Richards (2011) proposed a novel method to combine taxiway and runway scheduling elements in one optimization problem. They solved a mixed-integer linear programming model, and results demonstrated that the average taxi time could be reduced by half, compared to the First-Come-First-Served approach (FCFS). Yu et al. (2017) investigated gate reassignment and taxiway scheduling, simultaneously. They proposed a new heuristic approach to solve the integrated problem. Results highlighted their method's strength compared to a sequential method which solved gate reassignment and taxiway scheduling problem separately. For the first time, Sama et al. (2018) considered the integrated problem of scheduling ground and air operations in an airport maneuvering area. To address the problem, they presented an alternative MILP graph formulation aiming to minimize the maximum delay and three other objective functions, separately. They proposed a heuristic procedure based on FCFS sequence of landing/take-off operations as well, and showed the method's strength compared to the common policies.

To the best of our knowledge, there was no research in the literature that integrated GAP and RSP considering multiple taxiways. In the real world, however, a hub-airport usually possesses a complex taxi-network with a number of alternative paths. Therefore, taxiway selection is another crucial task for traffic controllers, which must be consistent with gate and runway assignment. In this paper, a multi-objective MILP model is proposed to consider GAP, RSP and taxiway selection, simultaneously. This model provides traffic controllers with an integrated plan for a set of departing aircrafts.

3. Problem Formulation

3.1. Problem Definition

This paper is supposed to plan departing aircrafts' movements in a Terminal Maneuvering Area (TMA). Needless to say, the more efficient the plan, the lower the flight delay and the higher the utilization rate of ground facilities.

First, each aircraft must park in the ramp area in front of a gate a few minutes before take-off to be loaded, refueled and boarded. In this procedure, long walking distance from gate to gate should be prevented by planners in order to reduce transit passengers' dissatisfaction. It is assumed that the minimum time required for each aircraft to occupy a gate is predefined, and any deviation from that would be undesirable. Once gate operations have been completed, the aircraft should push-back and enter the taxi-network. Traffic controllers prefer to navigate the airplane through the shortest ground route, but sometimes this path is occupied and the aircraft is diverted to a longer taxiway, which consequently leads to additional fuel and environmental costs. After a while, the aircraft reaches its assigned runway and take-off operation takes place. The most important goal for runway planners is to minimize take-off tardiness considering the minimum separation time between successive take-offs.

In this paper, an integrated mathematical model is proposed in order to cover all abovementioned aspects of the problem. This model can deal with a wide range of multi-gate, multi-taxiway and multi-runway problems. It is assumed that gates, runways and taxi routes for departures are exclusive and separate from arrivals. Furthermore, more than one aircraft cannot be assigned to a gate/runway/taxiway at the same time. Most parameters and variables which are used in this model can be found in previous studies related to GAP or RSP (Nourmohammadzadeh, 2012; Aktel et al., 2017; Guepet et al., 2016). However, some novel notions related to the variability of taxi routing and availability of multiple taxiways are also proposed for the first time.

3.2 Mathematical Model

In this section, a MILP model is developed to help airport planners optimally assign ground resources, i.e., gates, runways and taxiways. This model is an extension of the model presented by Nourmohammadzadeh (2012). He considered GAP and RSP with a fixed taxi time. To modify the model and make it closer to what happens in practice, the parameters of taxi routing problem are identified from the literature and added to the model (Organisation de l'aviation civile international, 2004).

The notations including indices, parameters, and decision variables are as follows:

Indices

$i.i'$: $Index\ for\ departing\ aircrafts$ $(i = 1.2. \dots . I)$

j: $Index\ for\ gates\ of\ departures$ $(j = 1. \dots . J)$

r: $Index\ for\ runways\ of\ departures$ $(r = 1.2. \dots . R)$

k: $Index\ for\ arrived\ flights/gates$ $(k = 1.2. \dots . K)$

p: $Index\ for\ departure\ taxiways$ $(p = 1.2. \dots . P)$

Parameters

sz_i: $Size\ of\ flight\ i$ *(1=Small, 2=Medium, 3= Large)*

pr_i: $Priority\ of\ flight\ i\ for\ the\ airport\ managrment$ *(1=Low,2=Average,3=High)*

$lbag_i$: $Minimum\ time\ required\ for\ flight\ i\ to\ spend\ at\ a\ gate$

pbt: $Average\ time\ needed\ for\ each\ aircraft\ to\ push\ back\ and\ clear\ ramp\ infront\ of\ a\ gate$

tto_i: $Target\ take\ off\ time\ for\ flight\ i$

$sr_{ii'}$: $Minimum\ separation\ time\ required\ between\ flight\ i\ and\ i'\ operations\ on\ the\ same\ runway$

$lbtt_{ip}$: $Minimum\ time\ for\ flight\ i\ to\ pass\ taxiway\ p$

$lbftt_i$: $Minimum\ possible\ taxi\ time\ for\ flight\ i$ $(\min_p lbtt_{ip})$

pto_i: $Time\ needed\ for\ flight\ i\ to\ speed\ up\ on\ a\ runway\ before\ take\ off$

np_{ki}: $Number\ of\ transit\ passengers\ between\ arrived\ flight\ k\ and\ departing\ flight\ i$

gd_{kj}: $Interior\ block\ distance\ between\ gate\ k\ and\ gate\ j$

cto_i: $Undesirable\ coeficient\ for\ every\ extra\ second\ that\ flight\ i\ might\ spend\ on\ taxiing\ more\ than\ lbftt_i$

cg: $Undesirable\ coeficient\ for\ every\ extra\ second\ that\ each\ aircraft\ might\ spend\ at\ a\ gate\ more\ than\ lbag_i$

M: $A\ very\ large\ positive\ number$

Decision Variables

gs_i: $Time\ when\ flight\ i\ starts\ to\ occupy\ a\ gate$

ag_i: $Time\ that\ flight\ i\ spends\ at\ a\ gate$

gf_i: $Time\ when\ flight\ i\ clears\ the\ gate$

tt_{ip}: $Time\ needed\ for\ flight\ i\ to\ pass\ taxiway\ p$

ftt_i: $Final\ taxi\ time\ for\ flight\ i$

sto_i: $Time\ when\ flight\ i\ enters\ its\ runway$

h_i: $Take\ off\ time\ for\ flight\ i$

z_{ij}: $Binary\ variable\ equals\ to\ 1\ if\ flight\ i\ is\ assigned\ to\ gate\ j; 0\ otherwise$

y_{ir}: $Binary\ variable\ equals\ to\ 1\ if\ flight\ i\ is\ assigned\ to\ runway\ r; 0\ otherwise$

x_{ip}: $Binary\ variable\ equals\ to\ 1\ if\ flight\ i\ is\ assigned\ to\ taxiway\ p; 0\ otherwise$

$\zeta_{ii'j}$: $Binary\ variable\ equals\ to\ 1\ if\ flight\ i'\ is\ assigned\ to\ gate\ j\ immediately\ after\ flight\ i; 0\ otherwise$

$\delta_{ii'r}$: $Binary\ variable\ equals\ to\ 1\ if\ flight\ i'\ is\ assigned\ to\ runway\ r\ immediately\ after\ flight\ i; 0\ otherwise$

$\gamma_{ii'p}$: Binary variable equals to 1 if flight i' is assigned to taxiway p immediately after flight i; 0 otherwise

Bi-objective GRP model

$$Min\ z_1 = cg\sum_i (ag_i - lbag_i) + \sum_i cto_i(ftt_i - lbftt_i) + \sum_i pr_i(h_i - tto_i) \quad (1)$$

$$Min\ z_2 = \sum_i\sum_j\sum_k pr_i np_{ki} gd_{kj} z_{ij} \quad (2)$$

$$\sum_j z_{ij} = 1 \quad \forall i \quad (3)$$

$$\sum_p x_{ip} = 1 \quad \forall i \quad (4)$$

$$\sum_r y_{ir} = 1 \quad \forall i \quad (5)$$

$$ag_i \geq lbag_i \quad \forall i \quad (6)$$

$$gf_i = gs_i + ag_i + pbt \quad \forall i \quad (7)$$

$$ftt_i \geq tt_{ip} - M(1 - x_{ip}) \quad \forall i.p \quad (8)$$

$$sto_i = gf_i + ftt_i \quad \forall i \quad (9)$$

$$h_i = sto_i + pto_i \quad \forall i \quad (10)$$

$$h_i \geq tto_i \quad \forall i \quad (11)$$

$$\sum_{i'.i' \neq i} \zeta_{ii'j} \leq z_{ij} \quad \forall i.j \quad (12)$$

$$\sum_{i'.i' \neq i} \zeta_{i'ij} \leq z_{ij} \quad \forall i.j \quad (13)$$

$$\sum_i \sum_{i'.i' \neq i} \zeta_{ii'j} \geq \sum_i z_{ij} - 1 \quad \forall j \quad (14)$$

$$\zeta_{ii'j} + \zeta_{i'ij} \leq 1 \quad \forall i.i'.j; i \neq i' \quad (15)$$

$$gs_{i'} \geq gf_i - M(1 - \zeta_{ii'j}) \quad \forall i.i'.j; i \neq i' \quad (16)$$

$$\sum_{i'.i' \neq i} \gamma_{ii'p} \leq x_{ip} \quad \forall i.p \quad (17)$$

$$\sum_{i'.i' \neq i} \gamma_{i'ip} \leq x_{ip} \quad \forall i.p \quad (18)$$

$$\sum_i \sum_{i'.i' \neq i} \gamma_{ii'p} \geq \sum_i x_{ip} - 1 \quad \forall p \quad (19)$$

$$\gamma_{ii'p} + \gamma_{i'ip} \leq 1 \quad \forall i.i'.p; i \neq i' \quad (20)$$

$$gf_{i'} \geq sto_i - M(1 - \gamma_{ii'p}) \quad \forall i.i'.p; i \neq i' \quad (21)$$

$$\sum_{i'.i' \neq i} \delta_{ii'r} \leq y_{ir} \quad \forall i.r \quad (22)$$

$$\sum_{i'.i' \neq i} \delta_{i'ir} \leq y_{ir} \quad \forall i.r \quad (23)$$

$$\sum_i \sum_{i'.i' \neq i} \delta_{ii'r} \geq \sum_i y_{ij} - 1 \quad \forall r \quad (24)$$

$$\delta_{ii'r} + \delta_{i'ir} \leq 1 \quad \forall i.i'.r; i \neq i' \quad (25)$$

$$sto_{i'} \geq h_i + sr_{ii'} - M(1 - \delta_{ii'r}) \quad \forall i.i'.r; i \neq i' \quad (26)$$

$$gs_i.ag_i.gf_i.sto_i.h_i.tt_{ip}.ftt_i \geq 0 \quad \forall i.p \quad (27)$$

$$z_{ij}.x_{ip}.y_{ir}.\zeta_{ii'j}.\gamma_{ii'p}.\delta_{ii'r} \in \{0.1\} \quad \forall i.i'.j.p.r \quad (28)$$

Objective (1) is to minimize any deviation from predefined schedule. Objective (2) is to minimize passenger dissatisfaction due to the transfer walking distance between gates. Constraint (3)-(5) indicate that each aircraft can be assigned to only one gate; one path and one runway, respectively. Constraint (6) ensures that an aircraft cannot release its gate sooner than the predefined time. Constraint (7) calculates the time each aircraft leaves the gate. Constraint (8) notices minimum taxi time which is possible for aircraft i considering its taxiway. Constraint (9) and (10) define the start and finish time for each aircraft's take-off. Constraint (11) guarantees that no flight takes off sooner than the pre-scheduled time. Constraints (12)-(15) determine aircrafts' sequence at gates. Constraint (16) depicts that no two flights are assigned to the same gate at the same time. Constraints (17)-(20) specify aircrafts' sequence on taxiways. Constraint (21) shows that it is impossible for two aircrafts to use the same taxi path at the same time. Constraints (22)-(25) specify aircrafts' sequence on runways. Constraint (26) ensures separation requirements for all pairs of take-offs. Constraints (27) and (28) define the domains of decision variables.

4. Solution Approach

In this section, a classic multi-objective optimization method is first presented to cope with the proposed model. Since this model is NP-hard, two meta-heuristic algorithms are also provided to solve large scale test problems.

4.1. Normalized weighted sum method

Objective functions in a Multi-Objective Optimization Problem (MOOP) generally contradict each other. So, it is impossible to obtain a single solution which optimizes all objectives at the same time. To deal with such problems, many classic methods have been introduced to transform a MOOP to a single objective problem. Normalized Weighted Sum Method (NWSM) is one of these methods which combines all objectives into one objective function using weighting coefficients. Therefore, the mathematical model of a minimizing problem can be transformed as follows.

$$Min \sum_{j=1}^{m} w_j \frac{f_j(x)}{sf_{j.0}(x)}$$

$$\begin{aligned} &\text{S.t:} \quad g_i(x) \le 0 \quad i = 1.2. \dots . n \\ &h_k(x) = 0 \quad k = 1.2. \dots . o \\ &x \in X \\ &w_j \in [0.1] \quad j = 1.2 \dots . m \quad and \quad \sum_{j=1}^{m} w_i = 1 \end{aligned} \tag{29}$$

where w_j is the weight assigned to the objective function f_j, and $sf_{j.0}$ is the normalizing factor which can be considered as f_j^{max} (the upper bound of the j^th objective function). It is worth mentioning that every single simulation run in which a specific combination of coefficients is used, usually contributes to a unique solution. Therefore, changing weights in consecutive iterations results in detecting different parts of an optimal Pareto front (Hwang and Masud, 2012; Kim and Weck, 2005; Ryu et al., 2009).

4.2. Meta-heuristics

On the contrary, Multi Objective Evolutionary Algorithms (MOEA) are directly used to find the Pareto solutions in a single simulation run. In this paper, we have utilized two population-based meta-heuristic algorithms, namely NSGA-II and MOGWO, for large-scale instances and compared them using performance measures of MOOP. According to the studies on transportation planning problems, NSGA-II is one of the most well-known and widely used meta-heuristic algorithms, and MOGWO is one of the most efficient algorithms which its outperformance has been reported in many studies (Maadanpour Safari et al., 2021). Since our solution representation and the fitness function utilized in both algorithms are exactly the same, they are first presented in the following sub-section.

4.2.1. Solution representation and fitness function

The structure that represents each individual of the population consists of 5 rows (see Figure 2). The number of genes in each row is equal to the number of departing flights. Three first rows determine gate, runway and path assigned to each departing flight, respectively. Each gene of these three vectors must be generated as an integer number in their feasible range. The fourth row refers to the priority of each aircraft to operate. Genes in this part are scaled to a discrete number in [1, nflight]. The last row of the solution structure represents waiting coefficient and specifies the time during which aircrafts occupy gates. Each gene in this part takes a value in [0,1], so ag_i is calculated as $ag_i = lbag_i + (ubag_i - lbag_i) \times cw_i$

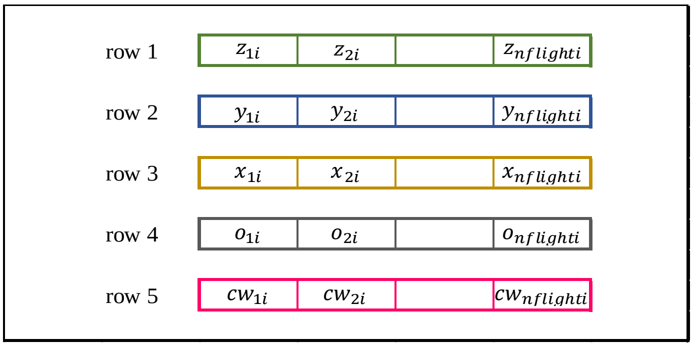

Fig. 2. Solution representation in meta-heuristic algorithms

Although this representation satisfies most constraints, it cannot fully guarantee the feasibility of a solution. With this representation, the problem of aircrafts' collision at gate/taxiway/runway still remains to be checked. To cope with this weakness, a penalty should be applied in fitness functions to penalize infeasible solutions based on the degree of violation. Therefore, in order not to assign aircrafts to the same resources at the same time, fitness function calculation for each individual is calculated as below.

1. Generate a random solution s_i

2. Flights assigned to the same gate are scheduled based on their priority one after another. Actually, an aircraft is authorized to park in front of a gate when the previous aircraft has spent the defined time and released the gate. If two flights have the same priority, the aircraft with lower flight number is first placed.

3. After each aircraft spends specific time in front of a gate, it is headed to a taxi path and then to a runway which are chosen for it.

4. Since some aircrafts have been directed to the same taxiway, it is crucial to determine whether the solution s_i violates taxi safety requirement. Therefore, as many times as concurrent usage of routes has been occurred, Deviation1 must be counted. The same procedure is applied for checking separation constraint on runways and Deviation 2's calculation.

5. Finally, the j[th] cost function for solution s_i is calculated as below, where W is the weight of deviations.

$$f_j(s_i) = \begin{cases} f_j(s_i) & \\ \quad if\ s_i \in feasible\ region & \\ f_j(s_i) + W \times (Deviation1 + Deviation2) & \\ \quad if\ s_i \notin feasible\ region & \end{cases}$$

4.2.2. NSGA-II algorithm

The Non-dominated Sorting Genetic Algorithm (NSGA) was first introduced in 1995 by Srinivas and Deb. the most important criticisms about this algorithm refer to 1) high computational complexity, 2) lack of elitism, and 3) need for identifying the shared parameter. So, Deb et al. (2002) added two operators to NSGA and introduced a new version called NSGA-II. These two operators are fast non-dominated sorting and crowding distance to check quality and diversity, respectively. In fact, solutions should be sorted based on non-domination rank (i_{rank}) and crowding distance ($i_{distance}$). NSGA-II procedure is explained step by step as follows:

1. Generate a random population of N chromosomes, P_0, and calculate the rank and crowding distance for every single solution
2. Set t=0
3. Implement the strategy of selecting parents, cross over and mutation on P_t to create offspring population, Q_t
4. Create R_t Merging P_t with Q_t
5. Implement non-dominated sorting on R_t to recreate a new population of size N, P_{t+1} (See Figure 3) and immediately calculate i_{rank} and $i_{distance}$ for all solutions in this new population.
6. If stopping criterion is satisfied, report all solutions in P_{t+1} with i_{rank}=1 as F_1, otherwise set t=t+1 and return to the step 3.

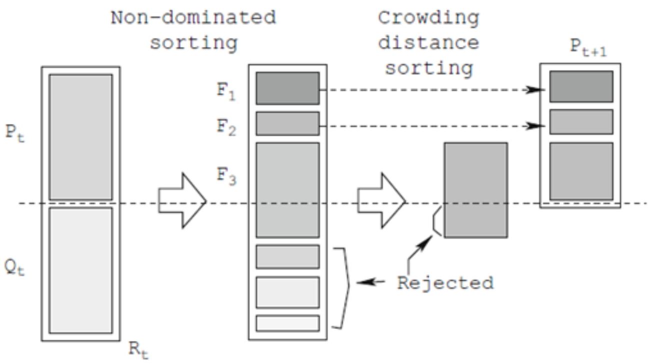

Fig. 3. Procedure of recreating a new population in NSGA-II (Deb et al., 2002)

In this paper, we assume that parents should be selected by binary tournament selection. Uniform and arithmetic crossovers are also designated for integer genes (rows 1 to 4) and continues genes (row 5) of chromosome (solution representation), respectively. The mechanism of uniform approach for row 1 (as the same way for rows 2, 3 and 4) is illustrated as follows.

$$\begin{cases} row\ 1\ of\ parent\ 1 = (z_{11}.z_{21}.\dots.z_{nflight1}) \\ row\ 1\ of\ parent\ 2 = (z_{12}.z_{22}.\dots.z_{nflight2}) \\ mask = (\alpha_1.\alpha_2.\dots.\alpha_{nflight}) \qquad mask\epsilon\{0.1\} \\ row\ 1\ of\ offspring1 = mask \times parent\ 1 + (1-mask) \times parent2 \\ row\ 1\ of\ offspring2 = mask \times parent\ 2 + (1-mask) \times parent1 \end{cases}$$

Note that there is a same procedure for row 5 except that $mask\epsilon[0.1]$.

In order to run mutation, some genes in each row should be selected and replaced by a new value in a feasible range. The procedure of the algorithm should be continued to reach maximum iteration as the stopping criteria.

4.2.3 MOGWO algorithm

Mirjalili et al. (2014) proposed Grey Wolf Optimizer (GWO) algorithm based on the mathematical model of grey wolves' behavior and their hunting method. Grey wolves generally live in the groups of 5 to 12 animals, and a social hierarchy is clearly seen in their group life. The wolf whose decision is a major determinant of hunting is the leader of the group and is called "Alpha wolf". "Beta wolf" is at the next level of the power pyramid. He helps Alpha wolf in his decisions, obeys him, and transmits Alpha's commands to the other wolves. The lowest level of the pyramid belongs to "Omega wolves" which are always surrendering to wolves with a higher social hierarchy. They play the role of scapegoats in hunting process and are the last wolves allowed to eat preys. If a wolf is not an Alpha, Beta or Omega, it is considered a "Delta wolf". The Delta wolves have to submit to Alpha and Beta but are superior to Omega wolves and also are considered as scouts, sentinels, elders, hunters, and caretakers of the group. In order to convert an optimization problem into a hunting process, we must consider the fittest solution as Alpha and the second and third best solutions as Beta and Delta, respectively; the rest of the candidate solutions are named Omega. In GWO algorithm, hunting (optimization) is guided by Alpha, Beta, and Delta wolves, while Omega wolves follow these three wolves.

Grey wolves encircle preys during the predation. Following equations are proposed to mathematically model encircling behavior in a hunt.

$$\vec{D} = |\vec{C}\cdot\vec{X}_p(t) - \vec{X}(t)| \tag{30}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}\cdot\vec{D} \tag{31}$$

$$\vec{A} = 2\vec{a}\cdot\vec{r}_1 - \vec{a} \tag{32}$$

$$\vec{C} = 2\cdot\vec{r}_2 \tag{33}$$

where t shows the current iteration and $\vec{A}$ and $\vec{C}$ stand for coefficient vectors. $\vec{X}_p(t)$ and $\vec{X}(t)$ are the position vectors of the prey and a gray wolf, respectively. Components of vector $\vec{a}$ linearly decrease from 2 to 0 over the consecutive iterations, and, $\vec{r}_1$ and $\vec{r}_2$ are random vectors in [0,1].

It is worth noting that in an optimization problem we have no idea about the location of the optimum solution (prey). Therefore, three best solutions have obtained so far are saved as Alpha, Beta and Delta and other search agents (Omegas) are obliged to update their positions according to the positions of the best solutions. Following equations are presented in this regard.

$$\vec{D}_\alpha = |\vec{C}_1\cdot\vec{X}_\alpha - \vec{X}|. \quad \vec{D}_\beta = |\vec{C}_2\cdot\vec{X}_\beta - \vec{X}|. \quad \vec{D}_\delta = |\vec{C}_3\cdot\vec{X}_\delta - \vec{X}| \tag{34}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1\cdot(\vec{D}_\alpha). \quad \vec{X}_2 = \vec{X}_\beta - \vec{A}_2\cdot(\vec{D}_\beta). \quad \vec{X}_3 = \vec{X}_\delta - \vec{A}_3\cdot(\vec{D}_\delta) \tag{35}$$

where $\vec{A}$ is a random value in the interval $[-2a.2a]$ and a decreases from 2 to 0 over the course of iterations. So, the fluctuation range of $\vec{A}$ is affected by $\vec{a}$. |A| < 1 forces wolves to attack towards the prey (exploitation) and |A| > 1 forces them to diverge from the prey in order to search for a fitter prey (exploration). $\vec{C}$ is another component which contains random values in [0,2] and provides random weights for the prey to emphasize exploitation ($C > 1$) or exploration ($C < 1$) . $\vec{C}$ helps the algorithm to attain a more random behavior not only during preliminary iterations, but also at all times.

In order to perform a multi-objective optimization by GWO, two new operators were considered by Mirjalili et al. (2016) and Multi-Objective Grey Wolf Optimizer (MOGWO) was presented. The first one is called "archive" which sorts non-dominated Pareto optimal solutions obtained so far. The second operator is a "leader selection strategy" to select Alpha, Beta and Delta from solutions in the archive in each iteration. In this paper, we consider the maximum number of iteration (MAXIT) as stopping criterion. The procedure of MOGWO is depicted in Figure 4.

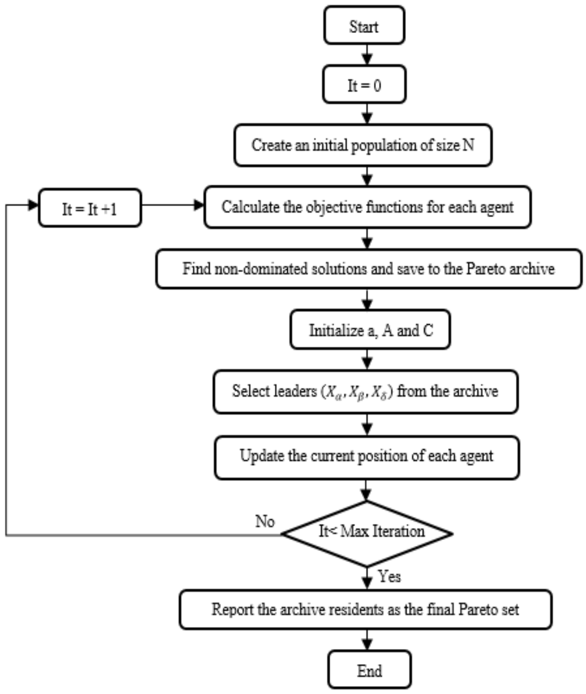

Fig. 4. Multi-objective Grey Wolf optimizer flowchart

5. Computational Results

In order to validate the suggested model, we consider the test problem No. 1 consisting of 7 flights, 3 gates, 2 paths, and 1 runway. The values of the model parameters have been generated according to Table 2. First, NWSM (using w1=0.6 and w2= 0.4) is considered to exert using GAMS 24.1.3 software on an Intel core™ i5, 1.6 GHz laptop with 4 GB RAM. The results represented in Table 3 validate that there is not any conflict in resource usage and model works properly. To attain the optimal Pareto front, various weighting vectors have been applied in NWSM. In addition, the assumed problem has been solved by NSGA-II and MOGWO in MATLAB R2013b and all obtained fronts is illustrated in Figure 5. As it can be seen, the metaheuristics generated fronts converged to the optimal Pareto front. Given that run time increases exponentially with the size of problems, NWSM is no more applicable for large-scale instances and the meta-heuristics are suggested to use.

Table 2
Input values of parameters in test problems

Parameters	Range	Parameters	Range
pr_i	Discrete.U[1,6]	$lbtt_{ip}$	U[120,480]
sz_i	Discrete.U[1,3]	pbt	U[60,120]
gd_{kj}	Discrete.U[1,10]	tto_i	U[1800,3600]
np_{ki}	Discrete.U[0,20]	cto_i	Discrete.U[1,6]
$lbag_i$	U[1500,2100]	cg	Discrete.U[1,6]
pto_i	U[10,30]		

Table 3
NWSM results using w1=0.6 and w2=0.4 for the problem No. 1

First objective	Second objective	Flight ID	Gate	Entering gate	Gate departing ~Taxi starting	Path	Taxi finishing ~Take-off starting	Runway	Taxi finishing
207.92	430	1	2	8:00:31	8:35:17	2	8:38:46	1	8:39:26
		2	2	8:35:17	9:09:29	2	9:12:31	1	9:13:21
		3	3	8:00:00	8:32:42	2	8:35:17	1	8:35:40
		4	1	8:31:54	9:00:53	2	9:04:36	2	9:05:15
		5	3	8:32:42	9:00:05	1	9:02:45	1	9:03:10
		6	3	9:00:05	9:28:52	1	9:30:48	1	9:32:05
		7	1	8:00:00	8:31:54	1	8:35:07	2	8:36:50

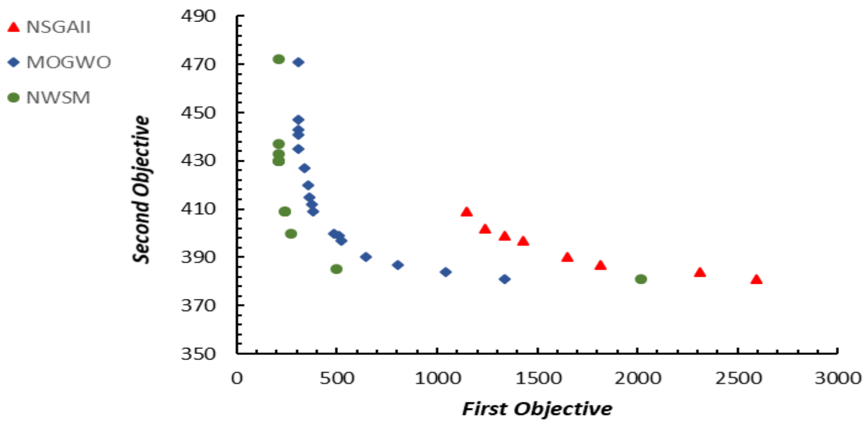

Fig. 5. NWSM comparison with NSGA-II and MOGWO on the problem No. 1

5.1. Performance measures

In the studies on MOOP, there are examples of efforts made to provide useful tools for assessing and comparing the performance of meta-heuristics. In this paper, three well-known measures are considered to quantify obtained Pareto fronts. The first criterion is Inverted Generational Distance (IGD) proposed by Sierra and Coello (2005) for measuring convergence. Note that the less IGD value, the better the performance.

$$IGD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \tag{36}$$

where n is the number of Pareto optimal solutions and $d_i = min_j \sqrt{(f_1^i(\vec{x}) - f_1^j(\vec{x}))^2 + \cdots + (f_m^i(\vec{x}) - f_m^j(\vec{x}))^2}$ $for\ all\ i.j = 1.2.\dots.n$. Note that m is the number of objective functions.

Next criteria are the spacing measure (SP) proposed by Coello et al. (2004) and maximum spread (MS) by Zitzler and Thiele (1999) to quantify the coverage of a Pareto set.

$$SP = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\bar{d} - d_i)^2} \tag{37}$$

where $d_i = min_j(|f_1^i(\vec{x}) - f_1^j(\vec{x})| + \cdots + |f_m^i(\vec{x}) - f_m^j(\vec{x})|$ $for\ all\ i.j = 1.2.\dots.n$, and $\bar{d}$ is the average of all d_is. It is worth mentioning that the lower value of SP shows the better performance of the algorithm.

$$MS = \sqrt{\sum_{i=1}^{m} \max(d(a_i.b_i))} \tag{38}$$

where $\max(d(a_i.b_i))$ is the Euclidean distance between the maximum and minimum values of the ith objective function. The more the maximum spread, the better the algorithm performance.

In order to generate more accurate Pareto fronts and have a fair comparison, the parameters of these algorithms should be tuned.

5.2. Parameter tuning

The performance of a meta-heuristic algorithm is to some extent sensitive to the setting of adjustable parameters affecting coverage and convergence behavior. In order to achieve the best performance of the proposed meta-heuristics, Taguchi's method is employed in this paper; L9 design for NSGA-II and L27 design for MOGWO. To attain more accuracy and precision, each experiment is repeated 5 times and the average of performance measures is considered. In the next step, performance measures (i.e., IGD, SP and MS) are normalized by related percentage deviation (RPD) formula. Then, a weighted linear combination of these measures is set as the main response. For both NSGA-II and MOGWO, parameters and their levels are given in Table 4. These levels were chosen based on the trial and error. Figures 6 and 7 show the results of Taguchi's method based on the signal to noise (S/N) ratio approach. For each factor, the optimal and appropriate level is the one with higher (S/N) value. It means that the larger value of S/N is better.

Table 4
Meta-heuristics parameters

Level	NSGA-II				MOGWO				
	Npop	Maxit	Pc	Pm	Archive size	Maxit	α	β	γ
1	40	100	0.7	0.1	40	100	0.06	2	1
2	70	150	0.8	0.15	70	150	0.1	4	2
3	100	200	0.9	0.2	100	200	0.14	6	3

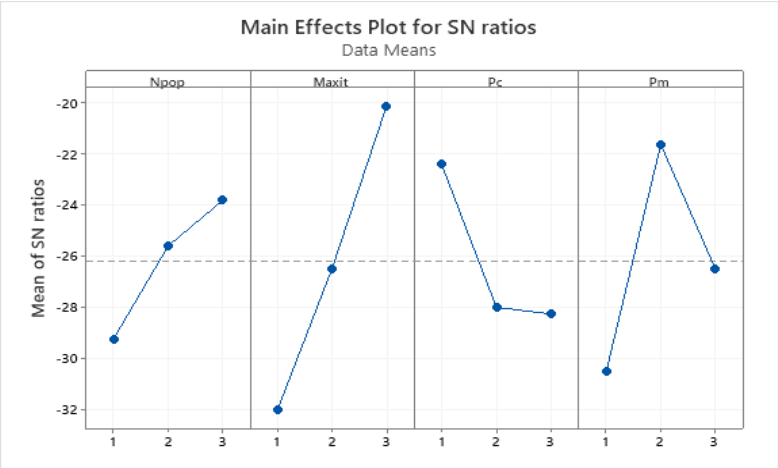

Fig. 6. S/N chart for NSGA-II parameters setting

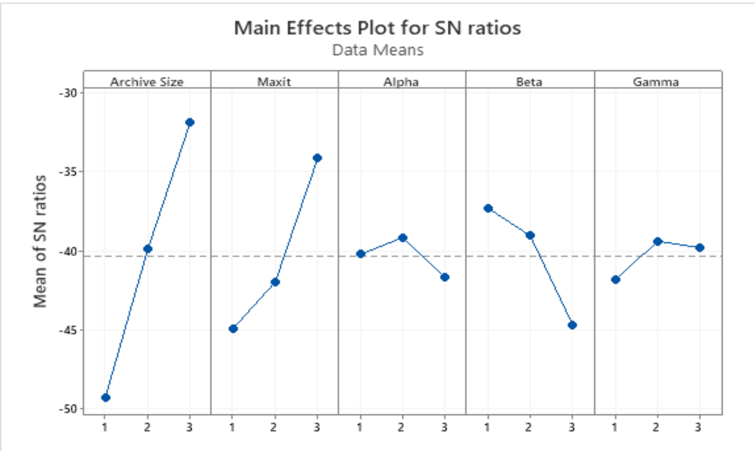

Fig. 7. S/N chart for MOGWO parameters setting

5.3. Algorithm analysis and comparisons

Since there was not any benchmark available in the literature, 25 test problems are designed according to Copenhagen (CPH) airport to analyze the performance of the presented meta-heuristics on large-scale problems as well. These generated instances are coded and carried out in MATLAB software using NSAGA-II and MOGWO algorithms. In all cases, the average results of 5 times run under three predefined performance criteria as well as CPU time are reported in Table 5. The characters in the column of problem feature from left to right present the number of flights, gates, paths and runways, respectively. For these cases, all metrics are plotted and graphically compared in Figures 8 and 9. As mentioned before, the lower the values of IGD and SP, the better the algorithm's performance. This is also true for CPU time criterion. Besides, we know that the more value of MS indicates the better quality of a solution set. As depicted in Figures 9 and 10, MOGWO in almost all cases generated lower value of IGD, SP and CPUT compared to NSGA-II. On the contrary, NSGA-II had a better performance in terms of MS. Broadly defined, MOGWO is superior to NSGA-II in coverage, convergence and time measures in our test problems. In order to assess the importance of the number of resources in an optimial planning, Pareto fronts of the test problems number 15 and 16 which have a remarkable difference in available resources are selected to be shown. As depicted in Figure 8, a greater number of gates, paths and runways contributes to a dramatic decrease in objective functions value. It is crucial for the airport management to make a trade-off between the cost of increasing resources and the cost of flight delays and passenger dissatisfaction. In addition, each of these Pareto optimal fronts illustrates a unique trade-off between the cost of schedule deviation and the undesirability of passenger walking distance. It is eventually up to the managerial viewpoint to select a plan from the Pareto optimal solutions. It depends on whether resource efficiency is a management priority or passenger dissatisfaction.

Table 5
Computational results of multi-objective metrics for NSGA-II and MOGWO

Problem NO	Feature	NSGA-II				MOGWO			
		IGD	SP	MS	CPUT	IGD	SP	MS	CPUT
1	6,3,4,1	28.33	28.28	17.2	196.02	13.56	23.11	11.43	72.72
2	6,3,3,2	30.16	16.23	14.78	199.34	14.38	28.08	11.49	118.82
3	6,3,3,1	28.54	12.59	14.33	193.82	14.36	28.01	11.49	115.02
4	6,2,2,2	67.26	65.69	33.59	188.67	23.37	53.61	27.63	141.25
5	6,3,2,1	36.57	17.07	16.01	188.01	14.12	27.27	11.40	152.36
6	6,2,2,1	61.06	84.23	34.155	188.03	23.37	53.61	27.63	143.611
7	7,3,3,2	62.85	90.71	45.33	203.18	30.69	65.66	32.05	117.05
8	7,3,3,1	32.71	69.06	45.97	205.63	28.94	58.23	33.02	105.83
9	7,2,2,2	48.57	98.36	41.89	202.89	34.33	78.57	35.38	127.28
10	7,2,3,1	84.20	135.53	44.35	204.91	35.67	76.83	35.69	126.13
11	7,4,3,1	68.47	182.04	59.57	206.23	30.9	89.1	47.33	96.44
12	7,3,2,2	66.04	133.52	52.32	206.86	33.4	83.03	42.37	119.27
13	8,4,1,2	76.42	114.97	55.22	194.73	41.18	63.83	40.89	137.12
14	8,4,4,2	96.42	142.53	55.18	191.81	40.44	99.03	41.61	113.09
15	8,4,4,1	92.11	135.42	55.36	191.18	43.03	91.41	43	101.44
16	8,8,8,2	56.66	203.2	77.63	191.02	23.4	67.78	41.94	115.66
17	9,5,5,1	45.03	136.97	63.03	193.46	27.79	69.68	44.86	104.52
18	9,9,9,3	32.82	122.37	58.38	196.52	14.71	54.9	38.5	96.77
19	9,9,9,1	32.4	98.95	57.44	203.74	27.25	66.25	41.73	105.01
20	10,10,2,1	30.7	112.28	72.68	200.34	18.56	44.98	39.46	108.12
21	10,10,5,2	58.32	202.03	66.84	202.83	18.63	65.76	37.04	107.82
22	10,10,10,1	23.59	99.07	59.76	202.19	36.45	145.97	52..28	95.5
23	15,10,10,l	44.84	217.03	109.73	222.21	27.39	107.64	55.05	126.19
24	15,15,15,1	22.97	91.75	92.27	225.67	33.9	91.21	53.66	119.29
25	15,15,15,2	47.25	206.22	93.37	230.83	148.68	618.50	80.25	128.55

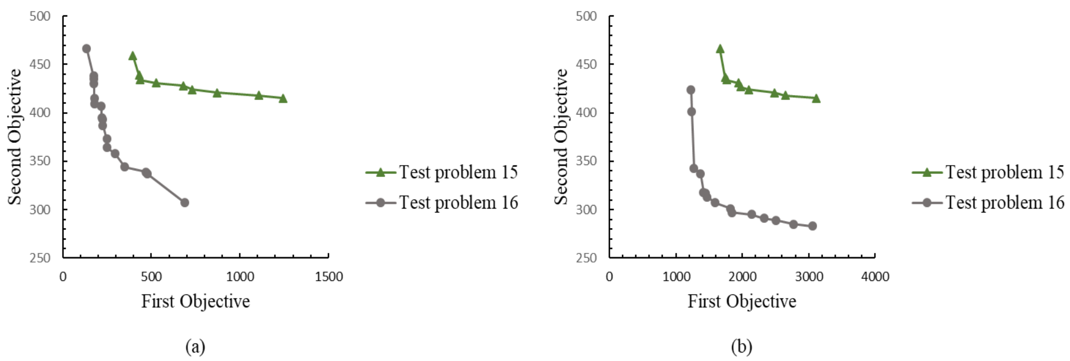

Fig. 8. The impact of resource number on objectives range using a) MOGWO and b) NSGA-II

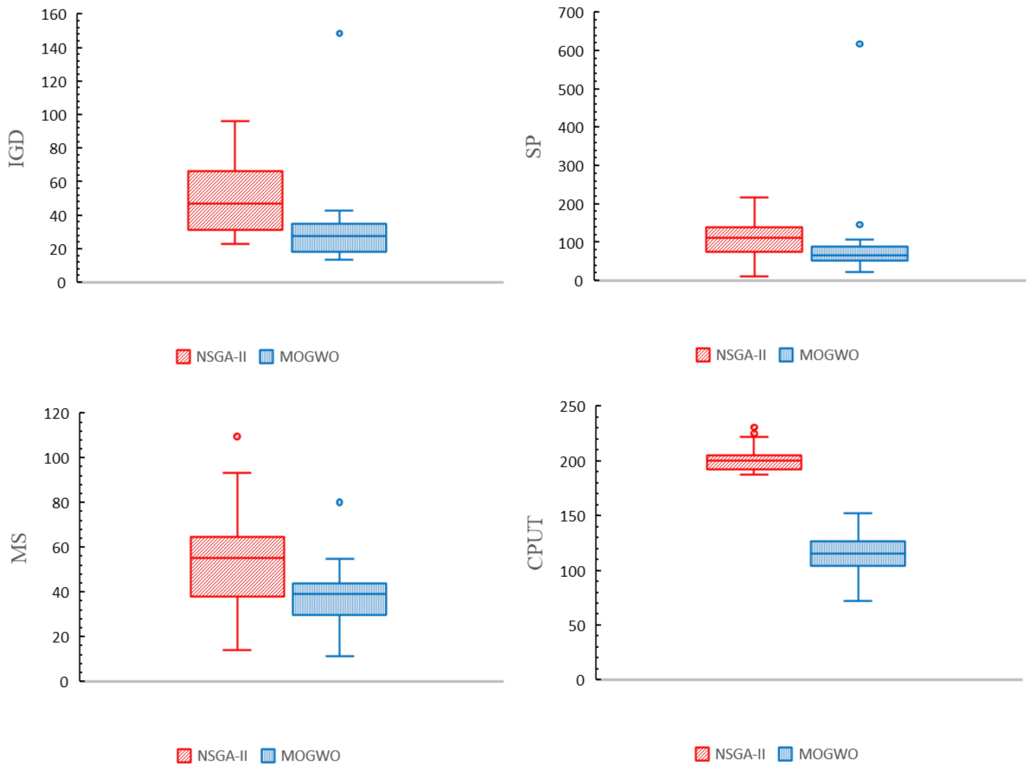

Fig. 9. Box-plot comparisons of meta-heuristics in terms of performance measures

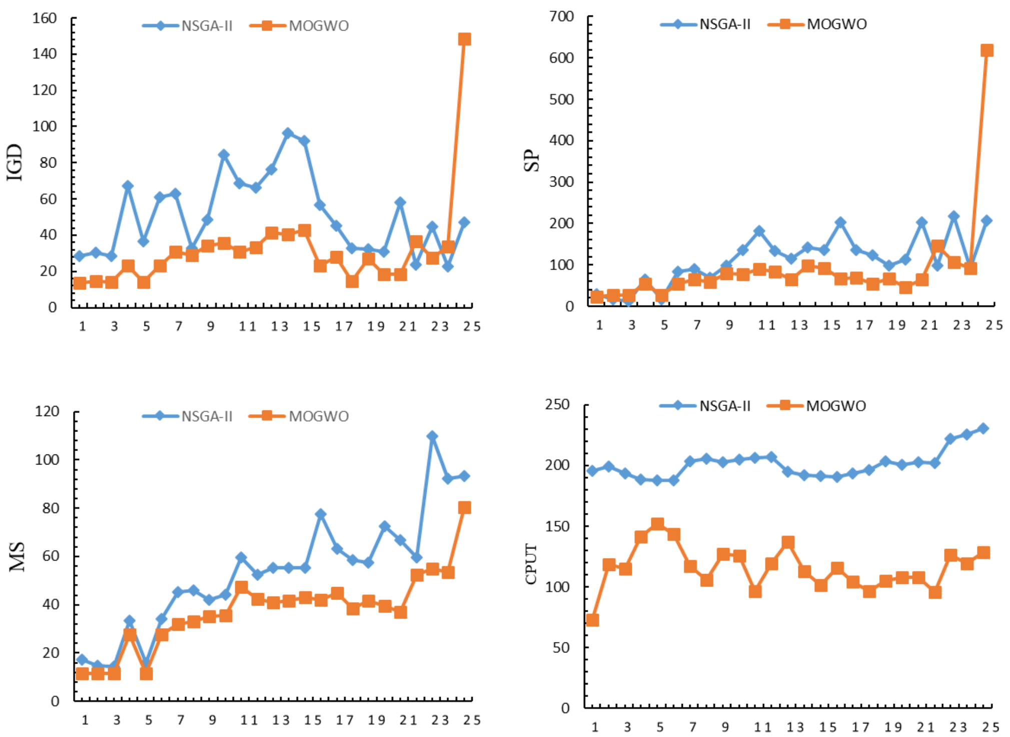

Fig. 10. NSGA-II and MOGWO performance on different test problems

6. Conclusion

In this paper, a new mathematical model is developed to integrate GAP and RSP considering alternative taxiways for departing aircrafts. The main contribution of this study is to take into account the crucial concept of taxi routing in airport planning as a key operation between gate and runway. A feasible taxi planning has a significant effect on preventing surface traffic at the airport and is considered as a preparation operation for takeoff. In our model, it is assumed that the airport network consists of multi-gates, multi-independent runways and also multi-heterogeneous taxiways. To meet international airport safety requirements, a sequence-dependent separation time is also considered between consecutive take-offs on the same runway. The proposed model has two conflicting objectives: 1) minimizing the cost of deviation from the preferred plan and 2) minimizing passenger walking distance between gates in TMA. The problem is first formulated in the framework of a constrained bi-objective mixed integer linear programming model. In order to find Pareto optimal fronts, the proposed bi-objective MILP is solved for small-sized problems by NWSM on Gams software. Given that the model developed in this study is computationally complex, two multi-objective evolutionary algorithms, namely NSGA-II and MOGW are employed to find Pareto fronts in large-sized test problems. Since the performance of meta-heuristic algorithms is sensitive to the setting of adjustable parameters, in order to achieve the best performance, Taguchi's designing method is utilized and parameters get fixed on their best levels. Results indicates the better performance of MOGWO compared to NSGA-II. MOGWO in almost all cases generates lower value of IGD, SP and CPUT in comparison to NSGA-II. However, NSGA-II shows a better performance in terms of MS. The impact of the number of resources on the costs related to delays and passenger dissatisfaction is also discussed. There are several recommendations for future works as follows:

- considering simultaneous arriving and departing aircrafts using shared resources
- considering domestic airline private resources in assigning gates to aircrafts
- considering WV effect on operations of different runways
- paying attention to the permitted level of noise pollution related to the aircraft operations at an airport near a residential area
- analyzing the problem under the uncertainty
- analyzing the optimal number of resources to design new airports according to the average number of flights
- developing the proposed model for an airport with multi-crossroads on taxiways

References

Aktel, A., Yagmahan, B., Özcan, T., Yenisey, M.M. and Sansarcı, E. (2017). The comparison of the metaheuristic algorithms performances on airport gate assignment problem. Transportation Research Procedia, 22, 469-478.

Atkin, J.A., Burke, E.K., Greenwood, J.S., and Reeson, D. (2007). Hybrid metaheuristics to aid runway scheduling at London Heathrow airport. Transportation Science, 41(1), 90-106.

Bennell, J.A., Mesgarpour, M., and Potts, C.N. (2011). Airport runway scheduling. 4OR, 9(2), 115.

Bennell, J.A., Mesgarpour, M., and Potts, C.N. (2017). Dynamic scheduling of aircraft landings. European Journal of Operational Research, 258(1), 315-327.

Clare, G., and Richards, A.G. (2011). Optimization of taxiway routing and runway scheduling. IEEE Transactions on Intelligent Transportation Systems, 12(4), 1000-1013.

Coello, C.A.C., Pulido, G.T., and Lechuga, M.S. (2004). Handling multiple objectives with particle swarm optimization. IEEE Transactions on evolutionary computation, 8(3), 256-279.

Daş, G.S., Gzara, F., and Stützle, T. (2020). A review on airport gate assignment problems: Single versus multi objective approaches. Omega, 92, 102146.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T.A.M.T. (2002). A fast and elitist multi objective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2), 182-197.

Guépet, J., Briant, O., Gayon, J.P., and Acuna-Agost, R. (2016). The aircraft ground routing problem: Analysis of industry punctuality indicators in a sustainable perspective. European Journal of Operational Research, 248(3), 827-839.

Hwang, C.L., and Masud, A.S.M. (2012). Multiple objective decision making—methods and applications: a state-of-the-art survey. Springer Science & Business Media, 164.

Khakzar Bafruei, M., Khatibi, S., & Rahmani, M. (2018). A Bi-Objective Airport Gate Scheduling with Controllable Processing Times Using Harmony Search and NSGA-II Algorithms. Journal of Optimization in Industrial Engineering, 11(1), 77-90.

Kim, I.Y., and de Weck, O.L. (2005). Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. Structural and multidisciplinary optimization, 29(2), 149-158.

Lee, J., Im, H., Kim, K. H., Xi, S., & Lee, C. (2016). AIRPORT GATE ASSIGNMENT FOR IMPROVING TERMINALS'INTERNAL GATE EFFICIENCY. International Journal of Industrial Engineering, 23(6).

Lieder, A., and Stolletz, R. (2016). Scheduling aircraft take-offs and landings on interdependent and heterogeneous runways. Transportation research part E: logistics and transportation review, 88, 167-188.

Maadanpour Safari, F., Etebari, F., & Pourghader Chobar, A. (2021). Modelling and optimization of a tri-objective Transportation-Location-Routing Problem considering route reliability: using MOGWO, MOPSO, MOWCA and NSGA-II. Journal of Optimization in Industrial Engineering, 14(2), 99-114.

Messaoud, M. B., Ghedira, K., & Harizi, R. (2017, October). The multiple runway aircraft landing problem: A case study for tunis carthage airport. In 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 2802-2807). IEEE.

Mirjalili, S., Mirjalili, S.M., and Lewis, A. (2014). Grey wolf optimizer. Advances in engineering software, 69, 46-61.

Mirjalili, S., Saremi, S., Mirjalili, S.M., and Coelho, L.D.S. (2016). Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. Expert Systems with Applications, 47, 106-119.

Nourmohammadzadeh, A. (2012). A comprehensive multi-objective mathematical model for the problem of landing and takeoff scheduling and gate assignment at the airport. Master Dissertation, Tehran University (UT), Tehran, Iran.

Organisation de l'aviation civile internationale. (2004). Advanced Surface Movement Guidance and Control Systems (A-SMGCS) Manual. ICAO.

Pohl, M., Kolisch, R., & Schiffer, M. (2020). Runway scheduling during winter operations. Omega, 102325.

Ryu, J.H., Kim, S., and Wan, H. (2009). Pareto front approximation with adaptive weighted sum method in multi objective simulation optimization. In Proceedings of the 2009 Winter Simulation Conference (WSC), 623-633.

Samà, M., D'Ariano, A., Corman, F., and Pacciarelli, D. (2018). Coordination of scheduling decisions in the management of airport airspace and taxiway operations. Transportation Research Part A: Policy and Practice, 114, 398-411.

Sierra, M.R., and Coello, C.A.C. (2005). Improving PSO-based multi-objective optimization using crowding, mutation and ∈-dominance. In International conference on evolutionary multi-criterion optimization, Springer, Berlin, Heidelberg, 505-519.

Yu, C. (2015). Airport ground services optimization with gate assignment and reassignment. Ph.D. Dissertation, The university of Hong Kong, Hong Kong.

Yu, C., Zhang, D., and Lau, H.H. (2017). A heuristic approach for solving an integrated gate reassignment and taxi scheduling problem. Journal of Air Transport Management, 62, 189-196.

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE transactions on Evolutionary Computation, 3(4), 257-271.

This article can be cited:
Larizadeh, R., Ramezanian, R. (2022). An Integrated Bi-Objective Mathematical Model for Minimizing Take-off Delay and Passenger Dissatisfaction. Journal of Optimization in Industrial Engineering, 15(1), 31-45.

http://www.qjie.ir/article_682268.html
DOI: 10.22094/joie.2021.1914607.1804